Real-Time News Detection and Analysis using Deep Learning and Large Language Models

---

Submission details:

1. Submission contains a zip file of repository and this readme file.
2. If there is an issue with zip file(due to size limits in e-learning) please clone the repository from:

   ```
   git clone https://github.com/harshalag21/media-bias-detector.git
   ```

3. Datasets are available at:

   ```
   https://github.com/harshalag21/media-bias-detector/tree/main/models/training/data
   ```

Execution Steps:

Codebase setup:

1. Clone repository:

   ```
   git clone https://github.com/harshalag21/media-bias-detector.git
   cd media-bias-detector
   ```

2. Install all dependencies:

   ```
   pip install -r requirements.txt
   ```

3. Config:
   - Verify NewsAPI key and Reddit credentials in `config/config.ini` ; Change if required
   - Change Elasticsearch password in `config/logstash-news-dash.conf`
   - Copy `config/logstash-news-dash.conf` to `$LOGSTASH_DIR/config`

Kafka and ELK setup:

1. Kafka
   - Create topic: news

     ```
     bin/kafka-topics.sh --create --topic news --bootstrap-server localhost:9092
     ```

   - Create topic: processed

     ```
     bin/kafka-topics.sh --create --topic processed --bootstrap-server localhost:9092
     ```

2. ELK Stack
   - Start Elasticsearch

     ```
     cd $ELASTICSEARCH_DIR; bin/elastic
     ```

   - Start Kibana

     ```
     cd $KIBANA_DIR; bin/kibana
     ```

- Start Logstash

```
cd $LOGSTASH_DIR; bin/logstash -f config/logstash-news-dash
```

- Create index: project

```
curl -X PUT "localhost:9200/project" -u user:password
```

Run code

1. Start spark code: processor.py

```
spark-submit --packages com.johnsnowlabs.nlp:spark-nlp-silicon_2.12:5.4.1,org.apache.spark:spark-sql-kafka-0-10_2.12:3.5.1 processor
```

Please wait till following line shows up:

```
Using an existing Spark session; only runtime SQL configurations will take effect.
```

2. Start news collector.py

```
python news_collector.py
```

Kibana dashboard

After about 40 seconds, the processed data should be visible in `Kibana->Analytics->Discover(select index "project")` for further visualizations. Sample dashboard designed is mentioned in reports.

Repository details:

File description:

1. Media Bias Dashboard - Elastic.pdf: Screenshot of dashboard
2. kafka_producer.py: python script for handling kafka connection
3. ner_analyser.py: pyspark script for extracting named entity count
4. news_collector.py: python script for fetching news from NEWSAPI and Reddit
5. processor.py: pyspark script for handling data processing and prediction
6. models/training: this directory has all the notebooks(colab) used for model training
7. models/training/data: this directory has all the training data

Directory structure

```
(.venv) ~/UTD/6350_BDA/media-bias-detector git:[main]
.
├── Media Bias Dashboard - Elastic.pdf
├── README.md
├── code_submission.txt
├── config
│         ├── config.ini
│         ├── logstash-news-dash.conf
│         └── parsedconfig.py
├── feed_csv.py
├── kafka_producer.py
├── models
│         ├── bias-detection
│         ├── category-detection
│         ├── sentiment-analysis
│         └── training
│                     ├── bias_detection.ipynb
│                     ├── data
│                     │       ├── category_detection_training.csv
│                     │       ├── news_category_test.csv
│                     │       ├── news_category_train.csv
│                     │       └── sentiment_analysis_training.csv
│                     ├── news_category_fine_tuning.ipynb
│                     ├── prediction.ipynb
│                     └── sentiment_analysis.ipynb
├── ner_analyser.py
├── news_collector.py
├── processor.py
├── requirements.txt
└── scraper
          ├── AllsidesDataScraper.ipynb
          └── data-preprocess.ipynb
```